TC5 - Audio Signal Declipping

You ZUO and Jingzhuo HUI

***Abstract*— This project aims to discuss the paper - Audio Declipping with Social Sparsity, and then implement the referred algorithm, a relaxed version of ISTA.**

I. INTRODUCTION

A. Audio Declipping Problem

An audio signal's clipping happens when some segments' amplitude exceeds the maximum value in a digital audio system. The clipping operation will thus truncate the exceeding parts to their corresponding limits. Audio declipping is the solution to the clipping problem; it helps to recover the missing or corrupted pieces of a signal. More precisely, taking into a voice signal normalized as an example (the blue signal in Figure 1.) . We set the clip rate $\theta_{clip} = 0.6$, which means that all the signal with its absolute values of amplitude superior to this rate will be truncated to 0.6 (the orange signal in Figure 2.).

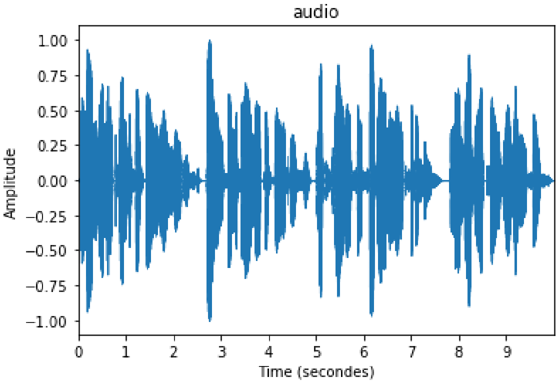

Fig. 1. orginal audio signal

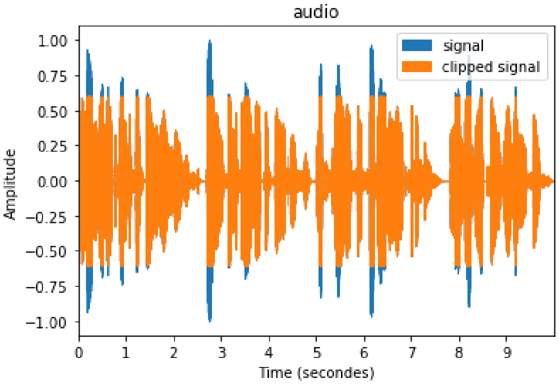

Fig. 2. orginal audio signal and its clipped signal

The problem is to reconstruct the original signal based on the truncated one, with the assumption that the original signal x can be represented sparsely with a Dictionary $\Phi \in \mathbb{C}^{T\times N}$ ($T \quad N$ is the number of) with corresponding synthesis coefficients $\alpha \in \mathbb{C}^N$.

Besides having an error term between the sparse representation and the clipped audio, the paper insists on adding an additional constraint to ensure that for the clipped segments, their reconstructed samples must be more significant (in absolute value) than the clipping threshold. Finally, they defined the problem of declipping as:

$$\begin{gathered}\hat{\alpha} = \arg\min_{\alpha} ||\alpha||_0 \\ s.t.\ \ ||y^r - M^r\Phi\alpha||_2^2 \leq \epsilon \quad and \quad |M^c\Phi\alpha| \geq |\theta^{clip}|\end{gathered} \tag{1}$$

where the matrix M^r (M^c) is comprised of matrices comprised of those rows of the identity matrix that choose the entries of the reliable (unreliable or clipped) samples.

Regarding the properties of the time-series signal, the author referred that the Gabor frames (Short-Time Fourier-Transform) have better performance than the other sparse decomposition methods, so they implemented the Gabor frames Φ during all the experiment process.

II. INNOVATION

Most of the previous works mentioned in the paper solved the declipping problems by using OMP (orthogonal matching pursuit) without the constraint $|M^c\Phi\alpha| \geq |\theta^{clip}|$, thus it leads to the classical Lasso problem:

$$\alpha = \arg\min_{\alpha} \frac{1}{2}||y^r - M^r\Phi\alpha||_2^2 + \lambda||\alpha||_1 \tag{2}$$

Like what we have done for the image inpainting problem, this convex non-smooth function can easily be minimized by the iterative shrinkage/thresholding algorithm or its accelerated version FISTA. But with the added constraint it can no longer be solved directly by this forward-backward algorithm. Moreover, there were some other algorithms proposed to use an inner iteration inside the forward-backward algorithm to approximate the l_1 penalty, or to figure out various structured sparsity approches. But they were all accompanied by a very high computational burden.

To solve these problems, authors of the paper proposed firstly an unconstrained convex relaxation of (1) by means of a squared hinge function, which is defined as follows:

$$h^2 : \mathbb{R} \longrightarrow \mathbb{R}_+ \quad z = \begin{cases} z^2 & if \ z < 0 \\ 0 & if \ z \geq 0 \end{cases} \tag{3}$$

They applied it as a penalty function, which takes $z = x - \theta^{clip}$ as inputs. Intuitively, when the unreliable parts x_k exceed the clipped value, it will raises the penalty, and 0 if not. Thus they reintroduced the unconstrained convex declipping problem in the following way, in which the form allowed to resolve it with ISTA-type algorithm.

$$\underset{\alpha}{\operatorname{argmin}} \frac{1}{2}||y^r - M^r \Phi\alpha||_2^2 + \frac{1}{2}[\theta^{clip} - M^c \Phi\alpha]_+^2 + \lambda||\alpha||_1 \tag{4}$$

Apart from this contribution, they also explored and compared some recently investigated social sparsity methods, such as Windowed Group-Lasso (WGL), the Empirical Wiener (EW) and the persistent EM (PEW).

- WGL: $\tilde{\alpha}_{tf} = \mathbb{S}_\lambda^L(\alpha_{tf}) = \alpha_{tf}(1 - \frac{\lambda}{|\alpha_{tf}|})^+$
- EW: $\tilde{\alpha}_{tf} = \mathbb{S}_\lambda^{EW}(\alpha_{tf}) = \alpha_{tf}(1 - \frac{\lambda^2}{|\alpha_{tf}|})^+$
- PEW: $\tilde{\alpha}_{tf} = \mathbb{S}_\lambda^{PEW}(\alpha_{tf}) = \alpha_{tf}(1 - \frac{\lambda^2}{\sum_{t' \in \mathcal{N}(t)} |\alpha_{t'f}|^2})^+$

where $\mathcal{N}(t)$ is the set of indices as the neighborhood of the index t for time-frequency coefficients $\alpha = \{\alpha_{tf}\}$, and $(x)^+ = \max(x, 0)$

The author articulated that the social sparsity allows incorporating the prior knowledge about signal classes and artifacts. The Group-Lasso techniques can be preferred to the Lasso since it provides a means for us to include (a certain type of) additional information into our estimate for the true coefficient. It has a group of coefficients to be jointly processed and consequently is more efficient and stable.

However, this kind of shrinkage operator typically leads to the loss of energy in the estimated signal. So in practice, people designed a new thresholding operator that preserves the energy in the significant coefficients, which is our EW defined as above. The author also explained that the EM operator features altered exponentiation of the coefficient energy while having the same support as the Lasso. As we can see, WGL is a persistent variation of Lasso, and the PEW is actually the persistent version of EM, which processes a bunch of coefficients jointly.

To rewrite the algorithm in detail:

Algorithm 1: relax version of ISTA

```
initialization: $\alpha^{(0)} \in \mathcal{C}^N$, $\alpha^{(0)} = z^{(0)}, \theta = ||\Phi\Phi^*||, k = 1$
while not convergence do
    g1 = $-\Phi^* {M^r}^T (y^r - M^r \Phi z^{(k-1)})$;
    g2 = $-\Phi^* {M^c}^T [\theta^{clip} - M^c \Phi z^{(k-1)}]_+$;
    $\alpha^{(k)} = \mathcal{S}_{\lambda/\delta}(z^{(k-1)} - \frac{1}{\delta}(g1 + g2))$;
    $z^{(k)} = \alpha^{(k)} + \gamma(\alpha^{(k)} - \alpha^{(k-1)})$;
    k=k+1;
end
```

III. Numerical Results

For the experiments, the paper evaluated the declipping perfoamance using the measure of SNR_m computed as:

$$SNR_m(y, \hat{y}) = 20\log_1 0 \frac{M^c y}{M^c(y - \hat{y})} \tag{5}$$

where y is the clipped signal and $\hat{y}$ its estimation

They used the "warm start" strategy to choose the threshold value λ; it starts at a relatively large value and decreases every step of iterations. This turns out to pick small values of λ, but since there is no noise in the clipped audio, we do not need large λ to fit it (the same conclusion as what we had in the inpainting problem). They set their hyperparameter space of λ from $\lambda = 10^{-1}$ to $\lambda = 10^{-4}$. By implementing their algorithm with $\theta^{clip} = 0.2$, they evaluated the SNR of different threshold methods with regards to time. We can see that PEW has the best SNR gain much better than the others. The second comes to the EW, the results of the others were not improved much, however. For the result of the amplitude curves, we can observe that PEW and EW are also more stable and fit better.

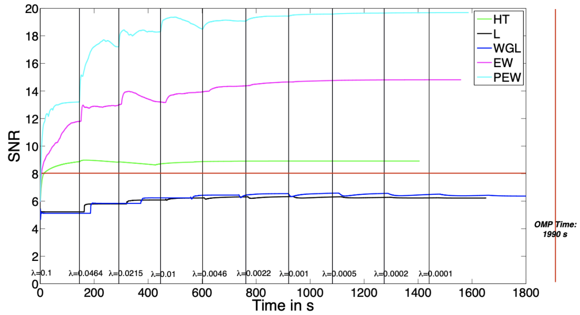

Fig. 3. Result of original paper[1

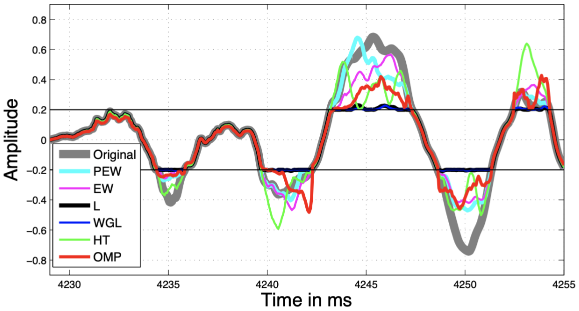

Fig. 4. Result of original paper[1

IV. Implementation of Algorithm

After carefully studying and analyzing the original paper, we also implemented this algorithm ourselves. We use a Gabor frame as a time-frequency dictionary, and the frame is based on a Hann window of 256 sample length (about 16 kHz audio sampling frequency) and a time-shift of 256 samples. And we set the relaxation coefficient $\gamma = 0.9$ and the clipping rate $\theta^{clip} = 0.6$(-0.6 for the negative amplitude segment) in our implementation. For the choice of λ, we simply tried with some small values and finally chose $\lambda = 0.01$

So as we applied the Short-Frequency Fourier Transform for our forward-backward computation, we have the results of the forward direction, the synthesis coefficients like:

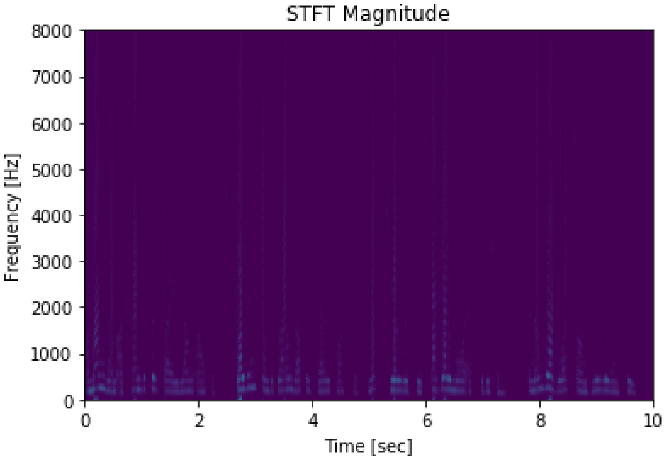

Fig. 5. stft

Then we apply a Soft Thresholding with threshold value $T = \frac{\lambda}{\delta}$, and do the inverse stft and get the result of the first iteration:

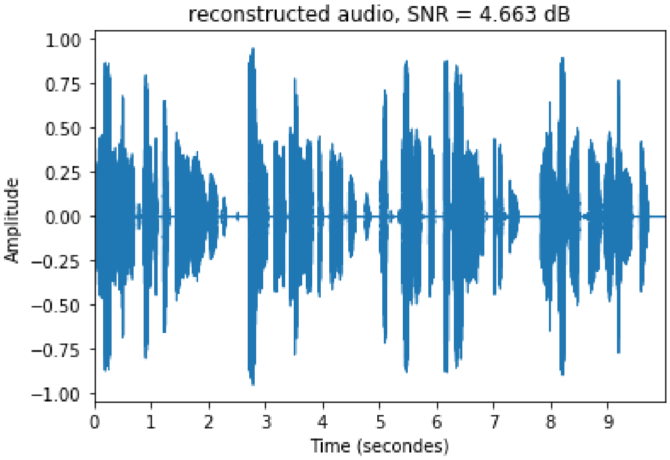

Fig. 6. istft after thresholding

Then after update the other parameters, we repeated the loop until convergence, and we get the reconstructed signal as:

As we can see, it only proved a tiny bit of the SNR, but it approximated more or less the original audio based on the clipped one.

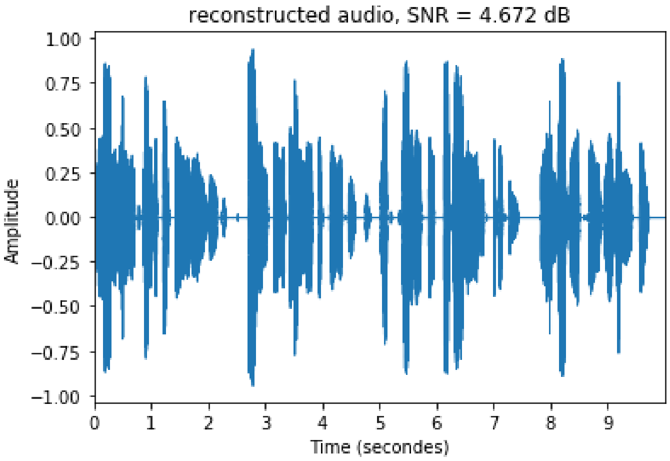

Fig. 7. Reconstructed audio

V. CONCLUSIONS

In this project, we first studied and made a discussion of the paper[1]. It introduced the fundamental problem of audio declipping and the classical ISTA algorithm to resolve it. Besides, it proposed a constraint that better expressed the properties of the truncated parts and turned it into an unconstrained convex declipping problem, which can be easily solved with an ISTA-type algorithm without high computational burden. The paper also demonstrated the effectiveness of different thresholding methods, like Lasso, WGL, EM, PEW, HT, etc.

Finally, we also implemented the algorithm a voice record lasting 10 seconds and successfully reconstructed it with SNR = 4.672 [dB].

REFERENCES

[1] Kai Siedenburg, Matthieu Kowalski, Monika Dörfler. Audio Declipping with Social Sparsity. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2014), May 2014, Florence, Italy. pp.AASP-L2. hal-01002998